processes

MDPI

Article

Real-Time Parameter Identification for Forging Machine Using Reinforcement Learning

Dapeng Zhang [1], Lifeng Du [2] and Zhiwei Gao [3,*]

[1] School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China; zdp@tju.edu.cn
[2] Tianforging Press Co., Ltd., Tianjin 300142, China; dlf@tjdy.com
[3] Faculty of Engineering and Environment, University of Northumbria, Newcastle upon Tyne NE2 8ST, UK
* Correspondence: zhiwei.gao@northumbria.ac.uk

Abstract: It is a challenge to identify the parameters of a mechanism model under real-time operating conditions disrupted by uncertain disturbances due to the deviation between the design requirement and the operational environment. In this paper, a novel approach based on reinforcement learning is proposed for forging machines to achieve the optimal model parameters by applying the raw data directly instead of observation window. This approach is an online parameter identification algorithm in one period without the need of the labelled samples as training database. It has an excellent ability against unknown distributed disturbances in a dynamic process, especially capable of adapting to a new process without historical data. The effectiveness of the algorithm is demonstrated and validated by a simulation of acquiring the parameter values of a forging machine.

Keywords: parameter acquisition; mechanism model; reinforcement learning; forging machine

Citation: Zhang, D.; Du, L.; Gao, Z. Real-Time Parameter Identification for Forging Machine Using Reinforcement Learning. *Processes* **2021**, *9*, 1848. https://doi.org/10.3390/pr9101848

Academic Editors: Rodolfo Haber and Jae-Yoon Jung

Received: 6 August 2021
Accepted: 14 October 2021
Published: 18 October 2021

1. Introduction

Complex engineering systems are with a high requirement for system reliability and control and production performance. A variety of technologies are developed to support the monitoring, optimization, and control for complex industrial processes such as chemical processes, manufacturing systems, power, and energy systems [1–3]. The forging process that enhances the mechanical properties by compressing the microstructure of parts [4] is widely applied in the fields of mining equipment, thermal hydro wind power generation equipment, nuclear power equipment, petroleum, and so on. As the key equipment, a forging machine should provide a precise pressing speed with a huge force to achieve the technological requirements of forging pieces. Therefore, the control of the forging machine is the guarantee of high forging quality. The control algorithms have made great progress from conventional PID-based algorithms [5] to advanced model-based control algorithms, including sliding mode control [6,7], back-stepping control [8], and feedback linearization [9], in order to obtain higher performance. However, the effects of these control algorithms strongly depend on the accuracy of the mechanism model. In [10,11], fuzzy-based control was proposed by using fuzzy rules instead of the mechanism model, but it cannot achieve the requirement of high precision. It is worthy to point out that the equivalent models, including regression models [12], neural networks [13], support vector machines [14], and so on [15], are alternatives of the mechanism model. These equivalent models overcome the difficulty of mechanical analysis, but at the cost of the model's extension and physical meanings. Up to now, the mechanism model is still feasible for precision control of the forging machine.

The mechanism knowledge of the forging machine has been mastered based on the related principles such as fluid mechanics, dynamics, and machinery technology. For example, the dynamic behaviors of the forging machine were analyzed according to the mechanism model [16]. A focus of the mechanism model with known structure is to determine the parameters, which is often by the way of offline identification and

online correction. Especially for a forging machine, most parameters come from the design handbook of forging machine [17] in which the values of parameters are recorded under the pre-set environment. The others are estimated based on the states of the forging machine by kinds of sensors. A number of offline identification methods such as least square method, maximum likelihood, Bayesian estimation, posteriori estimates, and minimizing maximum entropy were shown in reviews [18–20]. Reference [21] proposed to minimize the entropy of a kernel estimation, constructed from the residuals to deal with the case of not using the maximum likelihood estimation. In reference [22], a system parameter estimation method based on deconvolution of the system output process and explicit Levenberg optimization method was presented. Reference [23] presented a new derivative-free search method for finding models of acceptable data fit in a multidimensional parameter space and made use of the geometrical constructs known as Voronoi cells to derive the search in the parameter space. Reference [24] described a method for estimating the Nakagami distribution parameters by the moment method in which the distribution moments were replaced by their estimates. In order to trace the varying working parameters, the online estimated techniques were developed to improve the accuracy of model. The recursive parameter estimations were introduced to the linear model [25], the bilinear system [26], and the ARMA system [27]. In [28], an estimated noise transfer function was used to filter the input–output data of the Hammerstein system. By combining the key-term separation principle and the filtering theory, a recursive least squares algorithm and a filtering-based recursive least squares algorithm were addressed. Reference [29] proposed a parameter estimation algorithm using the simultaneous perturbation stochastic approximation (SPSA) to modify parameters with only two measurements of an evaluation function regardless of the dimension of the parameter. Reference [30] collected time-series data from an experimental paradigm involving repeated training and investigated the effect of various clustering methods on the parameter estimation. Reference [31] provided a servo press force by employing a novel dual-particle filter-based algorithm, achieving a maximum relative error in the force estimation of 3.6%.

As a foundation, a lot of effective historical data are necessary for parameter identification. Unfortunately, a forging machine is often working on batch processes whose parameters are different in each batch, and are even impossible to be known for new forging pieces. This means the parameters of the mechanism model for a forging machine will need to be determined from as few data as possible. From the perspective of data effectiveness, the classical parameter identification methods, whether offline estimation or online correction, are based on the least squares concept with the assumption of data following a normal distribution. It needs an appropriate window to observe the data because the statistical characteristics hide in the collected data. However, the difference of forging material quality and the variable pressure caused by pipe diameter change and flow rate change will lead to some disturbances that cause the data noise to be in an unknown distribution. So it is a challenge to determine the parameters of a model for a forging machine online to meet the needs of a complex environment.

Reinforcement learning (RL), motivated by psychology, statistics, neuroscience, and computer science, is about learning from interaction how to behave in order to achieve a design goal [32–34]. It will get rid of the limitation of training samples by learning directly from the raw data online. Through the learning process, an optimal action will be achieved to respond to the states. By sensing the current states, the RL does not need the assumption of prior distribution of noise. By episodes training, the action will overcome the overfitting difficulty and become robust due to eliminating the disturbance gradually. If the parameters were taken as the actions, they would be determined by reinforcement learning without thinking about the assumptions and disadvantages of the methods. In the case of a forging machine, it is a feasible approach to find the optimal values of the model parameters in a new condition under disturbances. There are some mature algorithms in the RL family, such as Q-learning [35], actor–critic [36], and deep reinforcement learning [37]. In this study, the Q-learning algorithm is proposed to determine the model parameters under the

working condition due to its simplicity. The contributions of this paper can be summarized as follows:

(1) The parameters are identified only based on the information of one period, which is promising for online control.
(2) The values of parameters are determined directly by raw data without any assumptions of noisy characteristics.
(3) The parameters have strong stability through a number of training episodes, which resists the bad influence of disturbance of unknown law.

The rest of this paper is organized as follows. Section 2 gives the model of pressing-down in forging machine that shows the state variables and the parameters. Section 3 describes the RL's procedure and releases the proposed approach. In Section 4, the model parameters are elaborated by the proposed approach and comparisons are made with two classical methods. Finally, conclusions are drawn in Section 5.

2. The Model of the Pressing-Down in Forging Machine

A semisolid metallic confectioning constant-speed isothermal forging is an important forging technique especially for light-weight alloy confectioning in the aerospace industry. The typical structure of the forging machine is illustrated in Figure 1, and the model has been built in our previous work [38]. It is repeated here for integrity.

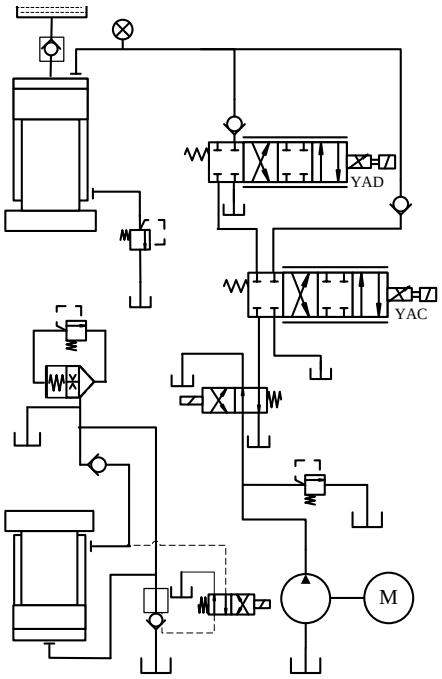

Figure 1. Typical structure of forging machine [38].

The function of the forging machine in pressing-down phase is affected by the oil pipe-line, the proportional servo valve, and the hydraulic cylinder with abandoning the auxiliary attachments.

2.1. The Oil Pipe-Line

The pressing speed in the pressing-down phase is always slow to meet the craft needs, so the oil works in the state of filament flow. Taking a pipe oil column as an object, the pressure balance equation is in the form of Formula (1).

$$\rho S_1 l \frac{d(q_1/S_1)}{dt} = (p_1 - p_s)S_1 - \frac{128\mu l}{\pi d^2} q_1 S_1 \quad (1)$$

Let R $= \frac{32\mu}{\rho}$, so Formula (1) becomes

$$\frac{1}{S_1}\frac{dq_1}{dt} = \frac{p_1 - p_s}{\rho l} + \frac{R}{S_1}q_1 \tag{2}$$

The difference between input volume and output volume is equal to the sum volume of oil compress and pipe swelling. So the oil continuity equation is

$$q_2 - q_1 = \frac{S_1 l}{K}\frac{d(p_1 - p_s)}{dt} \tag{3}$$

where q_1 and q_2 are the oil flow in pipe and the output oil flow of proportional servo valve, p_1 and p_s are the input pressure of proportional servo valve and the pressure of a constant rate pump output, S_1 and l are the sectional area of pipe and the length of oil pipe, and K is the young's modulus of oil equal volume.

2.2. Proportional Servo Valve

The proportional servo valve performs between the servo valve and the proportional valve. It eliminates the dead band by the way of fluid forerunner. The proportional servo valve is widely applied in the ultra-low-speed hydraulic machine to control the oil flow to the hydraulic cylinder. The proportional servo valve is described as

$$\frac{1}{\omega_n^2}\frac{d^2q_2}{dt^2} + \frac{2\xi}{\omega_n}\frac{dq_2}{dt} + q_2 = K_q A \tag{4}$$

where ξ amd ω_n are the damping rate and the inherent frequency of propositional servo valve, respectively, $K_q = K_n\sqrt{\frac{p_1 - p_2}{\Delta p_n}}$ is used to compensate the error between the practical pressure and criterion pressure, and A is the opening of proportional servo valve.

2.3. The Hydraulic Cylinder

The pipe-line between proportional servo valve and the hydraulic cylinder is omitted due to its short distance. The oil continuity equation of hydraulic cylinder is the form of

$$q_2 = S_2 v + \lambda_c p_2 + \frac{V_c}{K}\frac{dp_2}{dt} \tag{5}$$

where S_2 is the plunger's sectional area of exporting cavity of hydraulic cylinder, v is the moving speed of plunger, λ_c is the leak coefficient of hydraulic cylinder, p_2 is the output pressure of proportional servo valve, and V_c is the oil volume of upper cavity of hydraulic cylinder, $V_c = V_0 + vS$.

The dynamic equation of plunger is obtained according to the force analysis with the form of

$$p_2 S_2 + mg = m\frac{dv}{dt} + Bv + F + p_3 S_2 \tag{6}$$

where m is the mass of slider block, g is the acceleration of gravity, B is the viscous damping coefficient, F is the load resistance, and p_3 is the holding pressure of slide block. According to the design of forging machine, the holding power of slide block is equal to the gravity of slide block:

$$p_3 S_2 = mg \tag{7}$$

The Formula (6) is simplified to Formula (8) by substituting Formula (7) for Formula (6):

$$p_2 S_2 = m\frac{dv}{dt} + Bv + F \tag{8}$$

2.4. The Model of the System as a Whole

Let $x_1 = q_1,\ x_2 = p_1 - p_s, x_3 = \frac{dq_2}{dt}, x_4 = q_2, x_5 = p_2,$ and $x_6 = v$. By integrating the subsystems together, the global forging machine model can be described in the state–space form

$$\dot{x} = f(x) + g(x)u \tag{9}$$

where $x = [x_1, x_2, x_3, x_4, x_5, x_6]^T$, $u = A$,

$$f(x) = \begin{bmatrix} \frac{R}{S_1} & \frac{S_1}{\rho l} & 0 & 0 & 0 & 0 \\ -\frac{K}{S_1 l} & 0 & 0 & \frac{K}{S_1 l} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -2\xi\omega_n & -{\omega_n}^2 & 0 & 0 \\ 0 & 0 & 0 & \frac{K}{V_c} & -\frac{K\lambda_c}{V_c} & -\frac{KS_2}{V_c} \\ 0 & 0 & 0 & 0 & \frac{S_2}{m} & -\frac{B}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix},$$

$$g = [0, 0, 0, {\omega_n}^2 K_n \sqrt{\frac{x_2 - x_5 + P_s}{\Delta p_n}}, 0, -\frac{F}{B}]^T$$

Remark 1. *In the model, most parameters such as the length, the sectional area of oil pipe, the mass of slider block, and the rated flow gain can be valued according to the design. The values of parameters that are influenced by the surrounding or working conditions will result in the inaccuracy of model.*

3. The Proposed Method

3.1. Reinforcement Learning

The basic frame of reinforcement learning is shown in Figure 2. At each time step k, the agent makes observations $x(k) \in X$ and takes action $u(k) \in U$, and receives reward $R(x(k+1), x(k),\ u(k)) \in \mathbb{R}$.

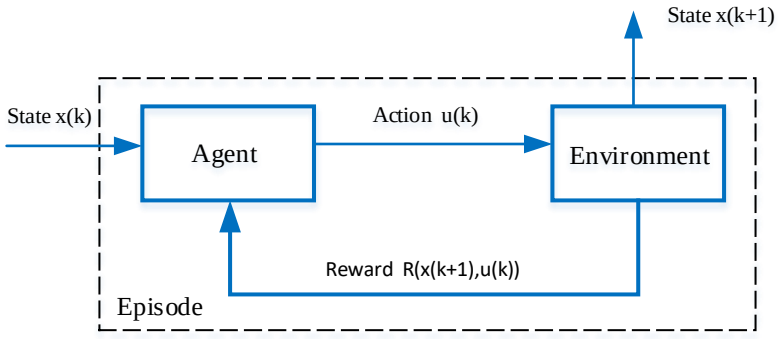

Figure 2. The basic frame of reinforcement learning.

The expected return that is received in the long run is described using the state–action value function $\mathrm{V}(x,\ u)$, under the condition of first taking an arbitrary action $u \in U$ from a certain state $x \in X$ and subsequently acting according to a certain control series π. So the value function $\mathrm{V}_\pi(x(k),\ u(k))$ at time k is defined as

$$\mathrm{V}_\pi(x(k),\ u(k)) = \sum_{t=k}^{\infty} \gamma R(x(k+1), x(t),\ u(t)) \tag{10}$$

where $\gamma \in [0, 1]$ is the discount factor.

The value function $\mathrm{V}_\pi(x(k+1),\ u(k))$ at time $k+1$ is defined as

$$\mathrm{V}_\pi(x(k+1),\ u(k)) = \sum_{t=k+1}^{\infty} \gamma R(x(t+1), x(t),\ u(t)) \tag{11}$$

According to the theory of dynamic programming

$$V_\pi(x(k),\ u(k)) = R(x(k+1), x(k),\ u(k)) + V_\pi(x(k+1),\ u(k)) \tag{12}$$

Unfortunately, the value function $V_\pi(x(k),\ u(k))$ and $V_\pi(x(k+1),\ u(k))$ is not obtained because no one knows the rewards after time $k+1$. To remove this obstacle, the Q-function is designed with $Q(x(k), u(k))$ and $Q(x(k+1), u(k))$ replacing $V_\pi(x(k),\ u(k))$ and $V_\pi(x(k+1),\ u(k))$, respectively

Let

$$\delta = R(x(k+1), x(k),\ u(k)) + \gamma Q(x(k+1), u(k)) - Q(x(k), u(k)) \tag{13}$$

The $u(k)$ will be optimized by a process of seeking δ approach to zero.

As an important member of reinforcement learning family, the basic step of Q-algorithm is carried out as Procedure 1 [30].

Procedure 1.

Initialize $Q(x(k), u(k))$ *arbitrarily*
Repeat (for each episode)
Initialize $x(k)$
Repeat (for each step of episode)
Choose $u(k)$ *from* $x(k)$ *using policy derived from Q (e.g.,* $\varepsilon - greedy$*)*
Take action $u(k)$*, observe* $R(k),\ x(k+1)$

$$Q(x(k), u(k)) \leftarrow Q(x(k), u(k)) + \alpha\left[R(k) + \gamma max_{u(k+1)} Q(x(k+1), u(k+1)) - Q(x(k), u(k))\right]$$

$x(k) \leftarrow x(k+1)$
until $x(k)$ *is terminal.*

Remark 2. *There is only state information in Procedure 1. One can obtain the optimal action online by using two states,* $x(k)$ and $x(k+1)$*, in the process of maximizing the value function. By this way, it makes an online control become possible because this approach gives up the requirement of sliding window length.*

3.2. The Proposed Approach

The scheme of proposed approach is shown in Figure 3.

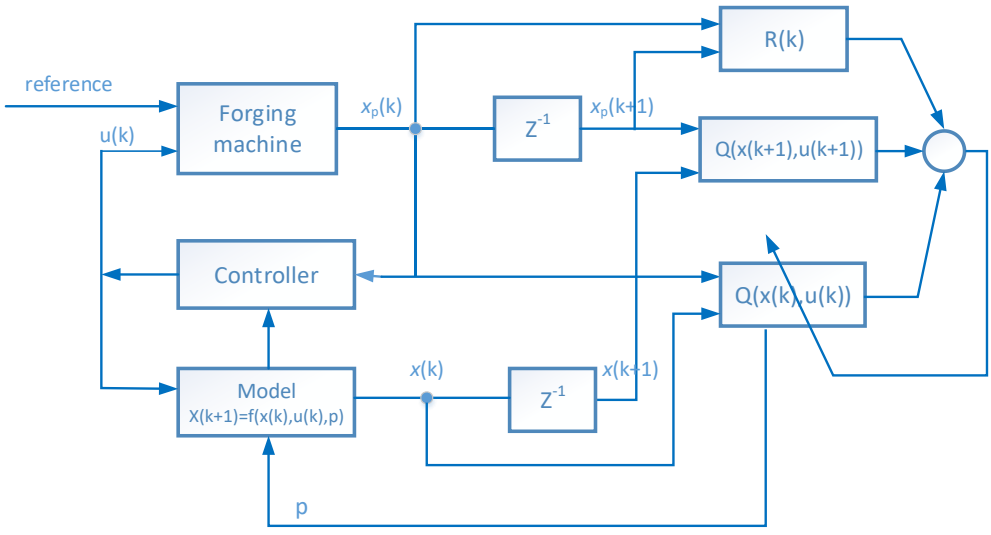

Figure 3. The scheme of proposed approach.

A model that consists of undetermined parameter p ($p \in R^m$) is paralleled to the forging machine under the controller. The state variables of model are recorded as $x(k)$ and $x(k+1)$ at sampling k and $k+1$, which are connected by a delay link z^{-1}. The unde-

termined parameter p is regarded as the action of Q-algorithm. Therefore, the Q-algorithm following Procedure 1 is applied to determine the parameter p based on $x(k)$ and $x(k+1)$ and finally, the optimal parameter p^* will be obtained when it is convergent.

To explicate Q-algorithm for the acquisition of model parameters, the key concepts of the proposed Q-algorithm are illustrated as follows.

(i). Action space, reward, and value function

The action space is made up of the undetermined parameter p. The values of parameter are usually inconsistent with the working condition, which will disturb with model accuracy. A goal is to determine their values responding to the surroundings.

The forging machine's velocity is designated a constant pressing speed or a given curve of speed during a certain temperature range according to the properties of forging materials, so the reward $\mathrm{R}(k)$ is selected as the reciprocal of change for absolution error between the measured speed and the set speed at adjacent sampling times k and $k+1$

$$\mathrm{R}(k) = \frac{1}{||v(k) - v_{set}(k)| - |v(k+1) - v_{set}(k+1)||} \tag{14}$$

where $v(k)$ and $v_{set}(k)$ are the measured speed and the preset speed at sample k; $v(k+1)$ and $v_{set}(k+1)$ are the measured speed and the preset speed at sample $k+1$. Here, using v instead of x_6 that is the sixth component of state vector x is only to stress the physics meaning.

Let $s = [x;\ u]$ so the value functions $\mathrm{V}(s(k), p(k))$ and $\mathrm{V}(s(k+1), p(k+1))$ from samples k and $k+1$ are defined by Formulas (15) and (16)

$$\mathrm{V}(s(k), p(k)) = \sum_{i=k}^{\infty} \mathrm{R}(i) \tag{15}$$

$$\mathrm{V}(s(k+1), p(k+1)) = \sum_{i=k+1}^{\infty} \mathrm{R}(i) \tag{16}$$

(ii). Q-function

The value functions $\mathrm{V}(s(k), p(k))$ and $\mathrm{V}(s(k+1), p(k+1))$ are replaced by Q-function according to the Q-algorithm because the value functions are not obtained due to the unknown rewards after sample k. The early Q-function that is applied for the discrete space is presented as a look-up table of states row and actions column. When the states or actions are continuous, their discretization will lead to the curse of dimensionality by generating an exponentially increasing complexity of algorithm and insufficient storage. Therefore, the parameterized function is proposed to fit the Q-function with the form

$$\mathrm{Q}(s(k), p(k)) = f(s(k), p(k), \theta) \tag{17}$$

where f and θ are a parameterized mapping and the parameters, respectively. Let $s = [s; p]$, an approximator is used to substitute for the unknown parameterized mapping, and there is

$$\hat{\mathrm{Q}}(s) = \sum_{i=1}^{\mathrm{n}} \phi_i(s)\theta_i \tag{18}$$

where $\phi_i(s, a)$ is usually selected as Gauss radial kernel function due to its simplicity, whose form is

$$\phi_i(s) = e^{-\frac{\|s - s_i\|^2}{2\sigma_i^2}} \tag{19}$$

in which s_i is the central coordinates of i-th radial kernel function and σ_i is the width of i-th radial kernel function.

(iii). Exploitation and exploration

There are two ways to determine the action in RL. The exploitation is used to get the best action from the Q-function that is based on the reward received. The exploration is used to escape the local optimization of exploitation by randomly giving the action. As a compromise of exploitation and exploration, the $\varepsilon-$ greedy algorithm is proposed to evolve the action. The agent selects the action that maximizes the Q-value function according to the probability ε that is usually a large probability event. In addition, it selects the action randomly according to the probability of $1-\varepsilon$ from the action space, which makes sure the action exploration is within the unknown area. The form of $\varepsilon-$ greedy algorithm is

$$p(k+1) = \begin{cases} \underset{p(k)}{argmax} Q(s(k), p(k), \theta), & Pr < \varepsilon \\ rand(U), & Pr \leq 1-\varepsilon \end{cases} \tag{20}$$

where $p(k)$ and $p(k+1)$ are the acquisition parameter at k and $k+1$, respectively, Pr is the probability of select action, and U is the action set.

(iv). The Process of Method

The proposed algorithm is summarized as Procedure 2. In this procedure, the input states are $x(k)$, $u(k)$ and $x(k+1)$, whose physical meanings are shown in Section 2, and the output parameter is p.

Procedure 2.

Step 1: Give a state $x(k)$ and the control $u(k)$ and then construct s according to $s = [x;\ u\ ]$

Step 2: Select parameters $p(k)$ randomly.

Step 3: Observe the next state $x(k+1)$

Step 4: Receive immediate reward $R(k)$ according to Formula (14)

Step 5: select $p(k+1)$ according to Formula (20)

Setp6: Compute $Q(s(k), p(k), \theta)$ and $Q(s(k+1), p(k+1), \theta)$ according to the Formulas (18) based on the model of Formula (9)

Step7: Compute the time series error $\delta(k)$ according to

$$\delta = R(k) + \gamma Q(s(k+1), p(k+1), \theta) - Q(s(k), p(k), \theta)$$

Step 8: Update $Q(s(k), p(k), \theta)$ according to

$$Q(s(k), p(k), \theta) \leftarrow Q(s(k), p(k), \theta) + \alpha\delta$$

Step9: $x(k) \leftarrow x(k+1), u(k) \leftarrow u(k+1)$ and $p(k) \leftarrow p(k+1)$

Step 10: Repeat steps 3 to 9 until it is convergent. The output p is the convergent $p(k)$ in which $p(k) = p(k+1) = p$.

(v). Convergence

The convergence of Q-algorithm can be found in [35,36].

4. Case Studies

The forging machine usually keeps a good state at the early life stage. In this stage, the values of parameters after a fine machine debugging always coincide with the design condition, except for the viscous damping coefficient B because it is prone to be influenced by the temperature and working condition. With time elapsing, the leakage becomes the main uncertainty of the forging machine. A little leakage is permitted for the forging machine if the leakage does not affect the work process. Nevertheless, the forging machine needs to be repaired if there appears much leakage. Therefore, we chose the viscous damping coefficient B and leakage coefficient λ_c as the identification parameters. These two parameters are unmeasurable, which make their values unverifiable in practice. As a result, we conducted a simulation to verify the proposed method.

4.1. Data Source

The state space model of (9) was used to simulate a forging machine. The values of model parameters are shown in Table 1 according to the design condition.

Table 1. The parameters values under the design condition.

ξ	ω_n	R	S_1	ρ	l	m	K	S_2	V_0	K_n	P_s	Δp_n
			m^2	kg/m^3	m	kg		m^2	m^2		Mpa	pa
0.7	70	0.0064	0.0138	870	7	1×10^4	1×10^{10}	0.02463	4.9×10^{-3}	2×10^{-4}	12	3.5×10^6

A controller is necessary for a forging machine to guarantee the quality of pressing process, therefore, a PID controller was used to simulate this situation. We chose a PID controller because here we focus on verifying our proposed method rather than discussing the control method. The PID controller is enough to provide the states and control for the proposed approach. The data series were generated by solving the model (9) with ODE45 that applies the fourth-order Runge Kutta algorithm to provide the candidate solution and the fifth-order Runge Kutta algorithm to control errors. These continuous sequences provided the data source by adding two kinds of noise with uniform distribution or Gaussian distribution as a simulation of real data. The set speed was changed from 0.02 to 0.08 that is consistent with the requirement of a typical pressing process. A typical control process that includes a transition process and a stable process is shown in Figure 4.

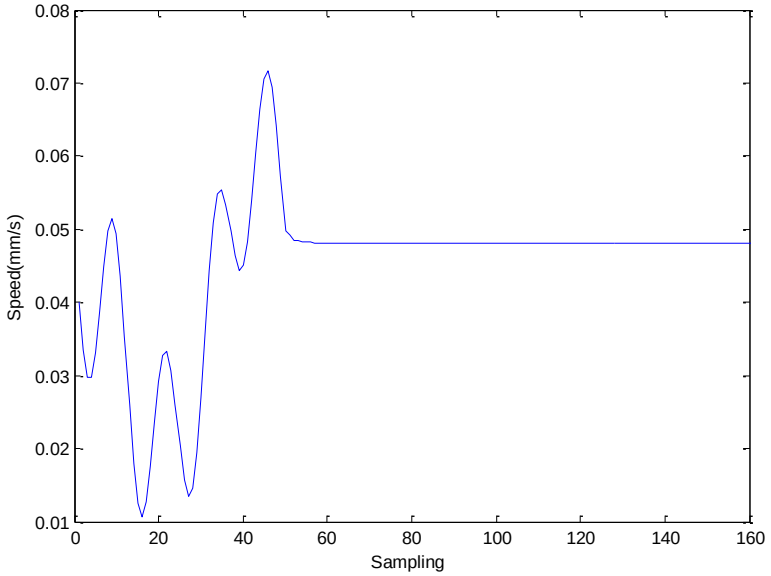

Figure 4. A typical control process (the set speed = 0.05).

The subsequent simulation was carried out at the platform of MatlabR2011b with the computer of Intel® Core™ 2 Duo CPU E7300 @2.66GHz 2.67GHz.

4.2. Acquisition of the Viscous Damping Coefficient

According to experiments, the viscous damping coefficient B is usually during 10–30 for this model. As a result, the value of 15 was chosen as the predetermined value and targeted by the proposed approach according to Procedure 2. The episodes training process is shown in Figure 5, where the subgraph above is with the noises of the uniform distributions and the subgraph below is with the noises of the Gaussian distributions. It is generally believed that the training time is related to the nature of the object and the computer performance. In order to avoid the time difference caused by different computer performance, we used the number of the episodes as an index of training time.

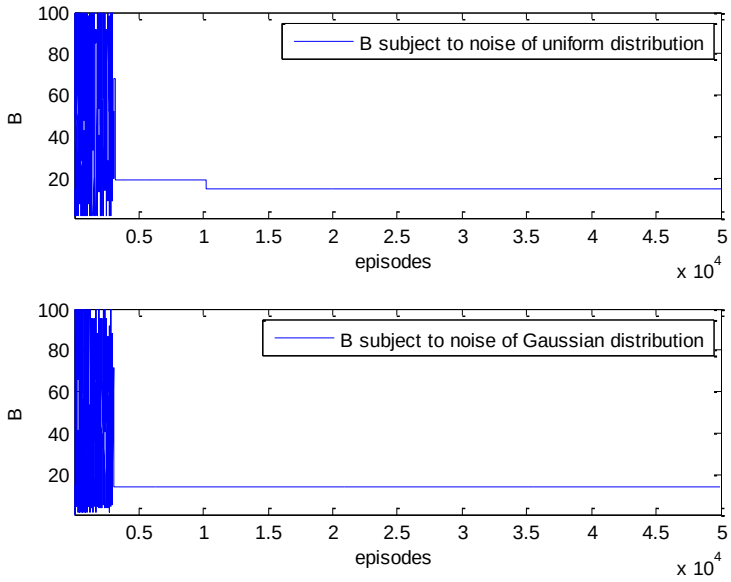

Figure 5. The episode training process of viscous damping coefficient (B was predetermined as 15).

Figure 5 shows there is a trial process at the beginning of training because there is no priori information on B. After a trial of about 3000 episodes, the best historical value of B that indicates 20 for the above subgraph and 15.0626 for the below subgraph appears during the process of seeking the best reward. After about 10,000 episodes, a better value of 14.5000 occurs for the above subgraph. In contrast, a value of 15.0626 for the below subgraph is unchanged until the episodes terminate.

The viscous damping coefficient B was changed from 15 to 20 to test the proposed method. The episodes training process is shown under a uniform distribution (the above subgraph) and under a Gaussian distribution (the below subgraph). Figure 6 shows the training episodes process similar to Figure 5. It is also seen that the trial process of Figure 6 lasts about 3000 episodes.

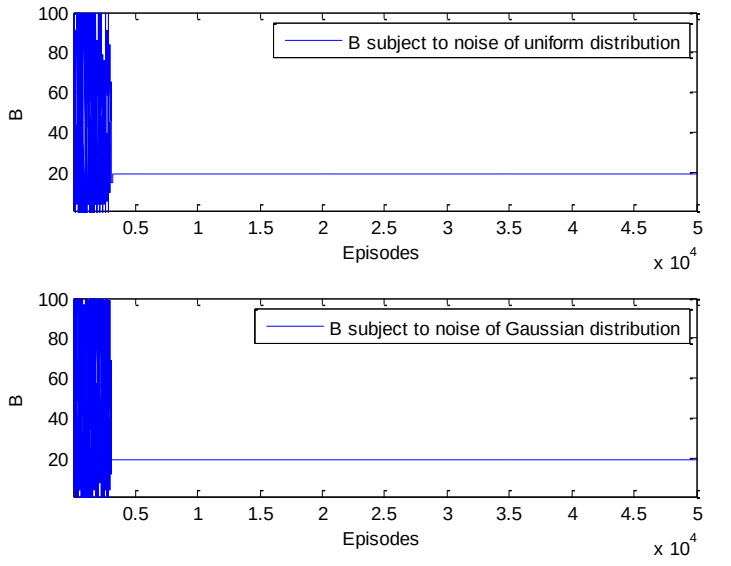

Figure 6. The episode training process of viscous damping coefficient (B was predetermined as 20).

In order to show the accuracy of parameter acquisition, the relative error δ between the estimated value $\hat{B}$ and the predetermined value B_r is defined as a form of

$$\delta = \left(\hat{B} - B_r\right) / B_r \tag{21}$$

and the results are shown in Table 2

Table 2. The results of viscous damping coefficient without leakage.

Predetermined Value	Noise Distribution	Acquisition	Relative Error
15	Uniform	15.0626	0.4%
15	Gaussian	14.5000	3.33%
20	Uniform	19.0000	5%
20	Gaussian	19.0000	5%

It is seen from Figures 5 and 6 that the excellent results with relative errors no greater than 5% were obtained in the cases of noises with different distributions.

Further tests under the condition of oil leakage were done to verify the effectiveness of the proposed approach. For a forging machine, the leakage is prone to go into saturation and is limited to a small value, so the leakage coefficients λ_c were assumed as a constant 0.01 and 0.02. The episodes training processes are shown in Figures 7–10. Figures 7 and 8 present the training processes of acquiring the viscous damping coefficient with a goal of 15 and of 20, respectively, under the leakage coefficient of 0.01. Figures 9 and 10 present the training processes of acquiring the viscous damping coefficient with a goal of 15 and of 20, respectively, under the leakage coefficient of 0.02. These figures show the proposed approach will be convergent after episodes training processes, and the final results are listed in Table 3. Table 3 shows the viscous damping coefficient will approach the predetermined value B_r under different coefficients or different noise distributions, showing a maximal relative error less than 2%. For training time, there are some differences for different parameters, such as about 6000 episodes in Figure 7, about 4000 episodes in Figure 9, and about 3000 episodes in Figure 10. Sometimes the different distributions also have an effect on the training speed, which is shown in Figure 8.

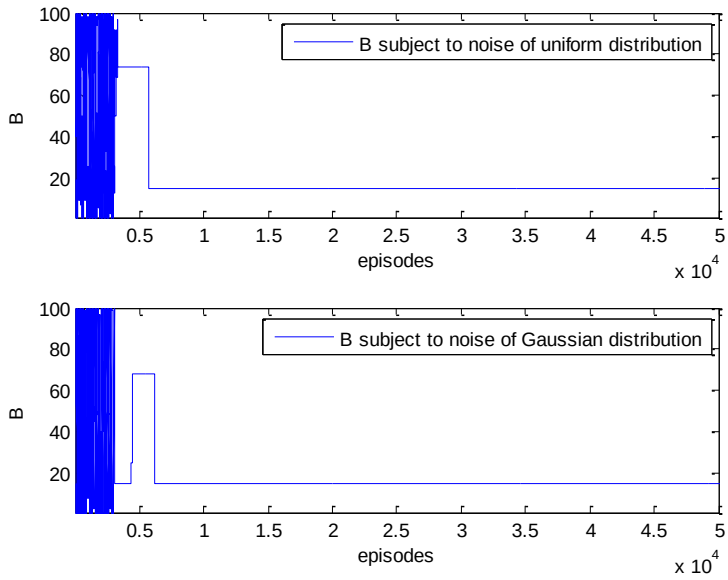

Figure 7. The episode training process with leakage of 0.01 (B_r = 15).

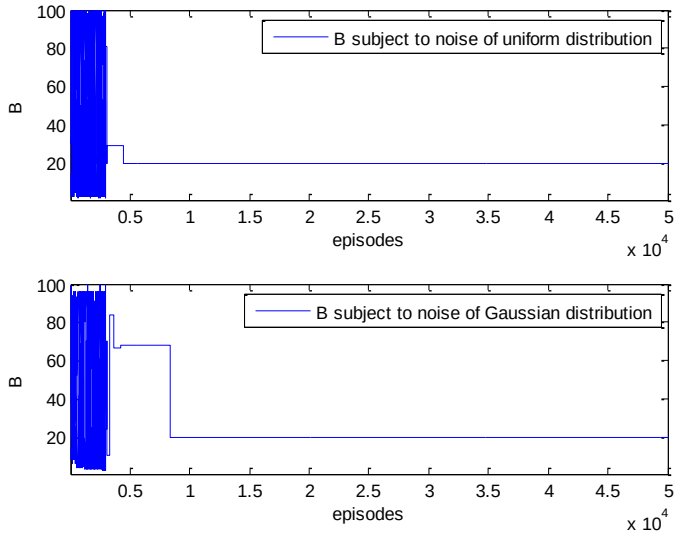

Figure 8. The episode training process with leakage of 0.01 (B_r = 20).

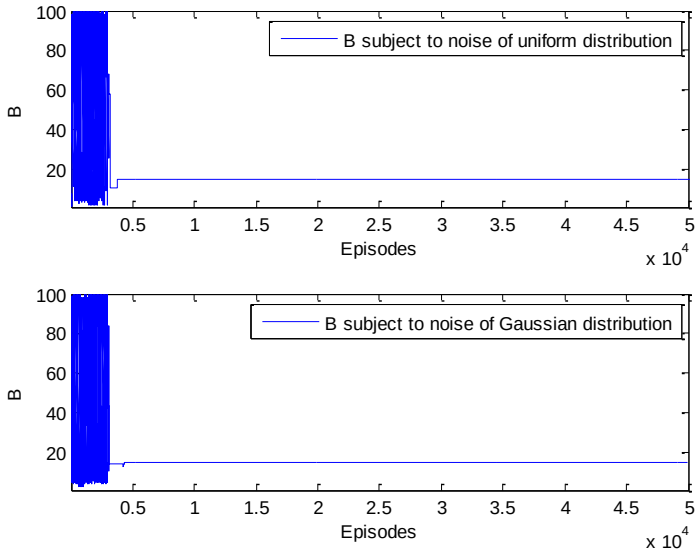

Figure 9. The episode training process with leakage of 0.02 (B_r = 15).

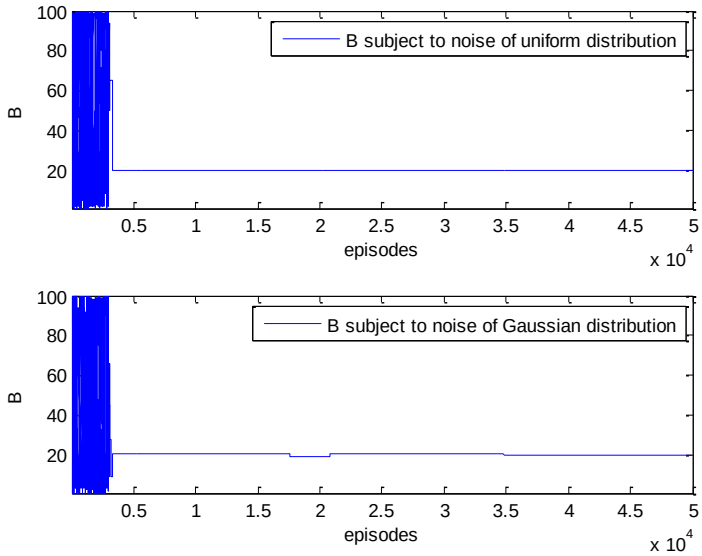

Figure 10. The episode training process with leakage of 0.02 (B_r = 20).

Table 3. The results of viscous damping coefficient under leakage.

Leakage Coefficient λ_c	Noise Distribution	Acquisition B	Predetermined Value B	Relative Error
0.01	Uniform	15.0000	15	0%
0.01	Gaussian	15.2500	15	1.67%
0.01	Uniform	20.0000	20	0%
0.01	Gaussian	20.0000	20	0%
0.02	Uniform	15.0000	15	0%
0.02	Gaussian	15.0000	15	0%
0.02	Uniform	20.0000	20	0%
0.02	Gaussian	19.9375	20	0.31%

4.3. Acquisition of the Leakage Coefficient

The leakage that is marked with leakage coefficient λ_c in the model will become the main uncertainty along with the lapsing time of forging machine. The leakage coefficient was predetermined as a constant 0.01 and 0.02. The learning processes with uniform distribution and with Gaussian distribution are shown in Figures 11 and 12, respectively. As for training time, it is affected by different distributions in Figure 11 and about 5000 episodes in Figure 12.

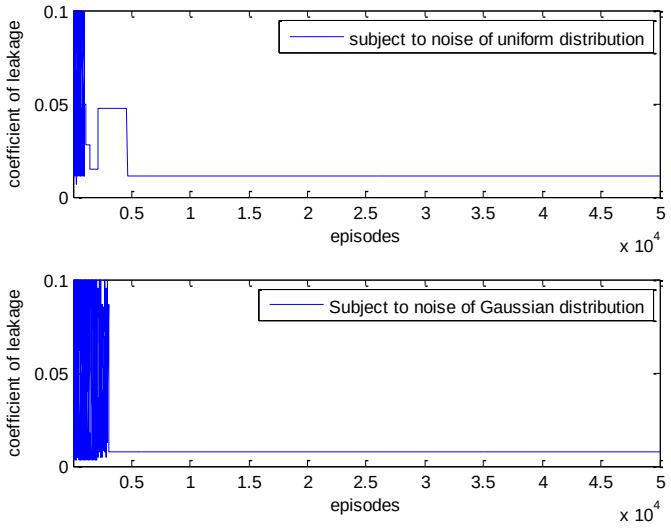

Figure 11. The learning process of leakage coefficient (λ_c was predetermined as 0.01).

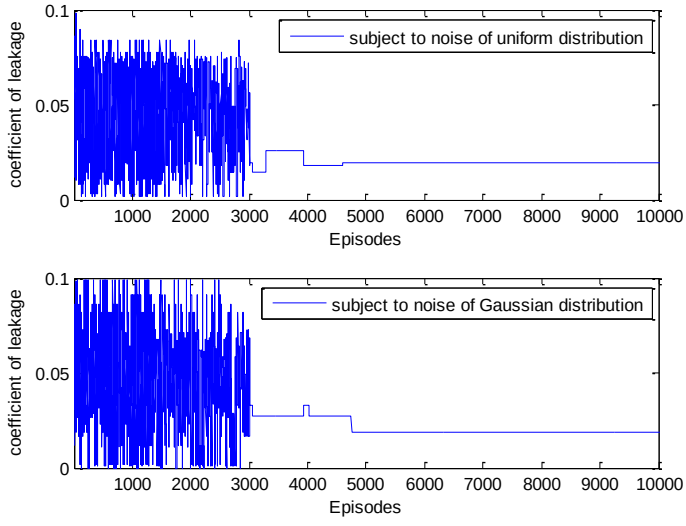

Figure 12. The episode training process of leakage coefficient (λ_c was predetermined as 0.02).

The values of leakage coefficient $\hat{\lambda}_c$ are acquired when the curve becomes stable. Here, the absolute error E with the definition of

$$E = \left|\lambda_c - \hat{\lambda}_c\right| \tag{22}$$

was used to replace the former relative error because the value of leakage coefficient is too small as the denominator of Formula (22), which is prone to an inappropriate relative error. The results are listed in Table 4. Table 4 shows the absolute errors are not more than 0.0015 in the cases of noisy with different distributions.

Table 4. The results of leakage coefficient.

Predetermined Value	Noise Distribution	Acquisition	Absolute Error
0.01	Uniform	0.0114	0.0014
0.01	Gaussian	0.0075	0.0015
0.02	Uniform	0.0200	0
0.02	Gaussian	0.0187	0.0013

4.4. Acquisition of the Viscous Damping Coefficient and the Leakage Coefficient

In order to test higher dimensionality of parameters, an experiment on acquiring concurrently the viscous damping coefficient and the leakage coefficient was done. The parameters of B and λ_c were predetermined as 18 and 0.01, respectively. The learning processes with uniform distribution and with Gaussian distribution are shown in Figures 13 and 14, respectively, and the results are shown in Table 5, which shows both parameters can reach a good estimation concurrently in the cases of noisy conditions. Here, all the training times are less than 5000 episodes.

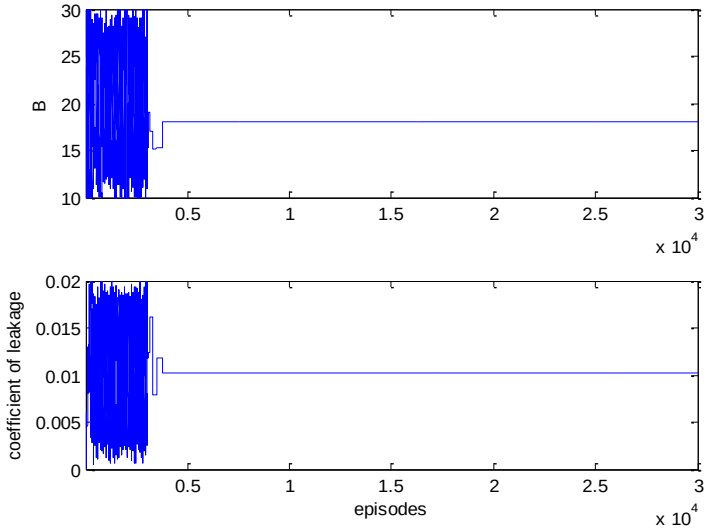

Figure 13. The episode training process of viscous damping coefficient and leakage coefficient subject to noise of uniform distribution.

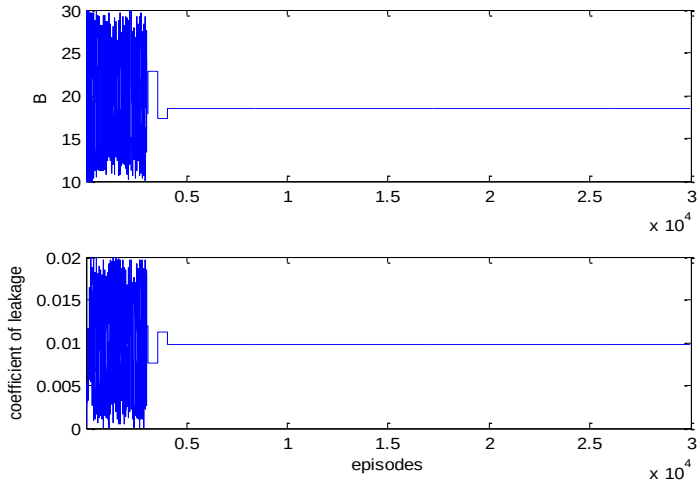

Figure 14. The episode training process of viscous damping coefficient and leakage coefficient subject to noise of Gaussian distribution.

Table 5. The results of the viscous damping coefficient and the leakage coefficient concurrently.

Noise Distribution	Parameters	Predetermined Value	Acquisition
Uniform	Viscous damping coefficient B	18	18.0488
	Leakage coefficient λ_c	0.01	0.0102
Gaussian	Viscous damping coefficient B	18	18.4141
	Leakage coefficient λ_c	0.01	0.0098

4.5. Comparison with Other Methods

A famous BP network approach and the sliding window correlation methods were chosen as a comparison of the proposed approach. The data series with 160 samples that was produced by the model with a controller was considered as the data source to determine the parameters. This data series includes a transient process of 50 and a stable process of 110 based on the viscous damping coefficient B of 15.

As we know, the BP network has a strong nonlinear approximation ability and an excellent estimation of recursion problem, which needs the length of input time series to match the order of the system. Here, we focused on identifying the parameter of viscous damping coefficient B just in one period. After several attempts, the BP network was chosen as a 7-20-1 structure with an input of seven variables (six states and one control in the model of Section 2) and an output of the viscous damping coefficient B. It was trained by the back propagation algorithm based on a train set of 2000 data from different cases in which the set speed was changed from 0.02 to 0.08. The learning rate was 0.001. The well-trained BP network was used to estimate the values of viscous damping coefficient, and the results are shown in Figure 15.

The values of viscous damping coefficient from sampling 1 to sampling 160 that were estimated by the BP network and the proposed approach are shown with the black curve and the red curve. It is seen that the BP network will approach to the viscous damping coefficient in the stable process, but it is bad in the transient process. The proposed approach shows an excellent performance that achieves the 15.0625 approaching to the goal of 15.0000 throughout the whole process.

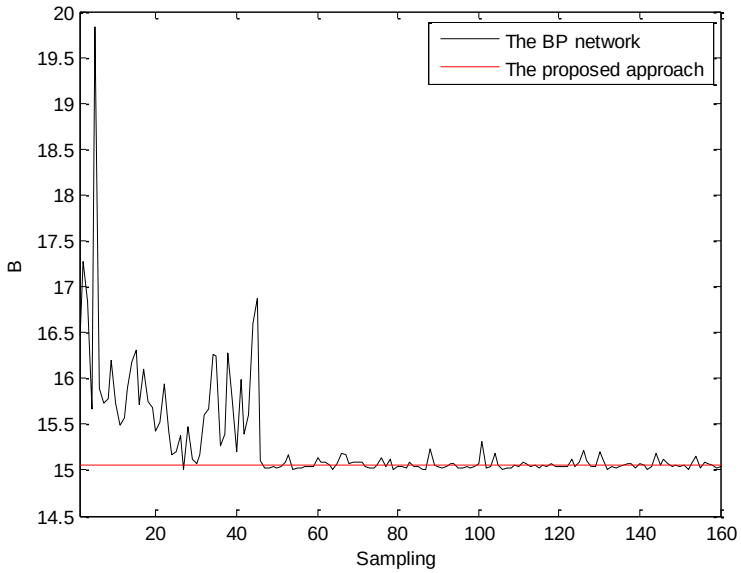

Figure 15. The comparison between BP network and the proposed approach.

The sliding window correlation method, as a kind of conventional parameters identification method for data series, was applied to estimate the values of viscous damping coefficient by an optimization of minimizing the sums of squared errors during each observation window. Considering the sliding window is influenced with the disturbance, it is prone to change the statistical properties of the observation window. The numbers of 2, 5, 10 and 50 were chosen as the length of sliding window, and the results are seen in Figure 16.

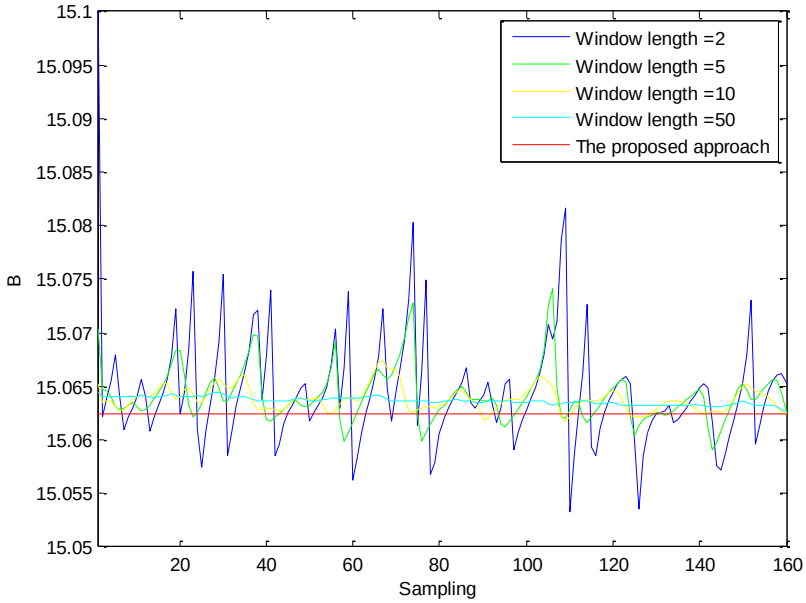

Figure 16. The comparison between the slide window and the proposed approach.

It is seen from Figure 16 that the sliding window correlation method and the proposed approach have a similar accuracy throughout the process from sampling 1 to sampling 160. However, there are some fluctuations for the sliding window correlation method according to different window length. The shorter the length of the slide window, the more sensitive the result, and vice versa. In contrast, the proposed approach shows a fine stability owing to its episodes training.

The advantages and disadvantages of three methods are summarized in Table 6.

Table 6. The comparisons of three methods.

	Advantage	Disadvantage
The BP networks	Learning algorithm, high accuracy in steady state	Worse in transient state
The sliding window correlation method	Optimization algorithm, high accuracy in steady state and transient state	Related to the length of the window and affected by disturbance
The proposed approach	high accuracy in steady state and transient state, only using the data during a period	Long training time

The proposed approach has the ability to obtain a high accuracy of viscous damping coefficient in steady state and transient state during only a period. To our best knowledge, there are no other approaches to implement the identification of model parameters with so little information, which is beneficial to the online control. However, it is limited to a slow process of the forging machine due to a long training time, though some improvements have been made, such as eligibility traces and heuristic search. A hardware implementation of this proposed approach is an attractive request for broader industrial processes.

5. Conclusions

In this paper, reinforcement learning has been addressed to identify optimal parameters values online by directly using raw data in one period. Compared with the BP network approach, the proposed technique has a good accuracy throughout the whole process. Compared with the sliding window correlation method, the proposed method has a similar accuracy but has a better ability to resist the influence of noise. As a result, the proposed approach has been demonstrated to be effective for online parameter identification in a simulation of real-time process of a forging machine.

Author Contributions: Conceptualization and methodology, D.Z. and Z.G.; formal analysis, L.D.; writing—original draft preparation, D.Z.; writing—review and editing, Z.G.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Program of Science and Technology Commissioner, and National Nature Science Foundation of China, grant number 61673074.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to acknowledge the research support from the School of Electrical Engineering and Automation at Tianjin University, and the E&E faculty at the University of Northumbria at Newcastle.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

Symbol	Meanings
ρ	Density of oil
μ	Dynamic viscosity
λ_c	Leak coefficient of hydraulic cylinder
ξ	Damping rate of propositional servo valve
ω_n	Inherent frequency of propositional servo valve
B	Viscous damping coefficient
d	Diameter of pipe
F_l	Load resistance
K	Young's modulus of oil equal volume
K_n	Rated flow gain

Symbol	Meanings
K_q	Flow gain of propositional servo valve
l	Length of oil pipe
m	Mass of slider block
p_1	Input pressure of proportional servo valve
p_2	Output pressure of proportional servo valve
P_s	Pressure of a constant rate pump output
Δp_n	Valve port pressure drop
q_1	Oil flow in pipe
q_2	Output oil flow of proportional servo valve
R	Intermediate coefficient
$S1$	Sectional area of pipe
$S2$	Plunger's sectional area of exporting cavity of hydraulic cylinder
u	Control voltage of proportional servo valve
v	Moving speed of plunger
V_0	Initial oil volume of upper cavity of hydraulic cylinder
V_c	Current oil volume of upper cavity of hydraulic cylinder

References

1. Gao, Z.; Chen, M.Z.Q.; Zhang, D. Special Issue on "Advances in condition monitoring, optimization and control for complex industrial processes". *Processes* **2021**, *9*, 664. [CrossRef]
2. Gao, Z.; Liu, X. An overview on fault diagnosis, prognosis and resilient control for wind turbine systems. *Processes* **2021**, *9*, 300. [CrossRef]
3. Gao, Z.; Dai, X.; Breikin, T.; Wang, H. Novel parameter identification by using a high-gain observer with application to a gas turbine engine. *IEEE Trans. Ind. Inform.* **2008**, *4*, 271–279. [CrossRef]
4. Available online: https://www.forging.org/producers-and-suppliers/technology/vision-of-the-future#importance (accessed on 1 May 2021).
5. Lu, X.; Huang, M. System-decomposition-based multilevel control for hydraulic press machine. *IEEE Trans. Ind. Electron.* **2012**, *59*, 1980–1987. [CrossRef]
6. Jia, C.; Wu, A.; Du, C.; Zhang, D. Variable structure control with sliding mode for a class of hydraulic nonlinear system. In Proceedings of the World Congress on Intelligent Control and Automation (WCICA), Jinan, China, 7–9 July 2010.
7. Ho, T.; Ahn, K. Speed control of a hydraulic pressure coupling drive using an adaptive fuzzy sliding-mode control. *IEEE-ASME Trans. Mechatron.* **2012**, *17*, 976–986. [CrossRef]
8. Li, C.; Wu, A.; Du, C. Speed control of hydraulic press via adaptive back-stepping. *Appl. Mech. Mater.* **2011**, *40–41*, 46–51. [CrossRef]
9. Zhang, D.; Wu, A.; Zhang, G.; Du, C. Application of the differential geometric feedback linearization to the speed control of forging machine. In Proceedings of the 2010 Chinese Control and Decision Conference, Xuzhou, China, 26–28 May 2010.
10. Lee, Y.; Kopp, R. Application of fuzzy control for a hydraulic forging machine. *Fuzzy Sets Syst.* **2001**, *118*, 99–108. [CrossRef]
11. Duan, X.; Deng, H.; Li, H. A saturation-based tuning method for fuzzy PID controller. *IEEE Trans. Ind. Electron.* **2013**, *60*, 5177–5185. [CrossRef]
12. Azari, A.; Poursina, M.; Poursina, D. Radial forging force prediction through MR, ANN, and ANFIS models. *Neural Comput. Appl.* **2014**, *25*, 849–858. [CrossRef]
13. Bharti, P.S. Process modelling of electric discharge machining by back propagation and radial basis function neural network. *J. Inf. Optim. Sci.* **2019**, *40*, 263–2778. [CrossRef]
14. Fan, B.; Lu, X.; Huang, M. A novel LS-SVM control for unknown nonlinear systems with application to complex forging process. *J. Cent. South Univ.* **2017**, *24*, 2524–2531. [CrossRef]
15. Hong, J.; Yeh, W. Application of response surface methodology to establish friction model of upset forging. *Adv. Mech. Eng.* **2018**, *10*, 1–9. [CrossRef]
16. Pan, Q.; Li, Y.; Huang, M. Estimation of dynamic behaviors of hydraulic forging press machine in slow-motion manufacturing process. *Nonlinear Dyn.* **2019**, *96*, 339–362. [CrossRef]
17. China Society for Technology of Plasticity CMES. *Forging Manual*; China Machine Press: Beijing, China, 2013.
18. Koksal, G.; Batmaz, I.; Murat, C. A review of data mining applications for quality improvement in manufacturing industry. *Expert Syst. Appl.* **2011**, *38*, 13448–13467. [CrossRef]
19. Olivier, A.; Smyth, A.W. Review of nonlinear filtering for SHM with an exploration of novel higher-order Kalman filtering algorithms for uncertainty quantification. *J. Eng. Mech.* **2017**, *143*, 04017128. [CrossRef]
20. Lu, X.; Huang, M. Novel multi-level modeling method for complex forging processes on hydraulic press machines. *Int. J. Adv. Manuf. Technol.* **2015**, *79*, 1869–1880. [CrossRef]

21. Pronzato, L.; Thierry, E. Entropy minimization for parameter estimation problems with unknown distribution of the output noise. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Saltlake City, UT, USA, 7–11 May 2001.
22. Slivinskas, V.; Simonyte, V. Modeling of a mechanical system using output data of the hammer blow sequence response. *J. Vibroeng.* **2009**, *11*, 120–129.
23. Ratnaweera, A.; Halgamuge, S.K.; Watson, H.C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. *IEEE Trans. Evol. Comput.* **2004**, *8*, 240–255. [CrossRef]
24. Artyushenko, V.M.; Volovach, V.I. Nakagami distribution parameters comparatively estimated by the moment and maximum likelihood methods. *Optoelectron. Instrum. Data Process.* **2019**, *55*, 237–242. [CrossRef]
25. Ram, S.S.; Veeravalli, V.V.; Nedic, A. Distributed and Recursive parameter estimation in parametrized linear state-space models. *IEEE Trans. Autom. Control* **2010**, *55*, 488–492. [CrossRef]
26. Zhang, X.; Ding, F. Recursive parameter estimation and its convergence for bilinear systems. *IET Control Theory Appl.* **2020**, *14*, 677–688. [CrossRef]
27. Chen, F.; Ding, F.; Sheng, J. Maximum likelihood based recursive parameter estimation for controlled autoregressive ARMA systems using the data filtering technique. *J. Frankl. Inst.* **2015**, *352*, 5882–5896. [CrossRef]
28. Wang, Z.; Shen, Y.; Ji, Z. Filtering based recursive least squares algorithm for Hammerstein FIR-MA systems. *Nonlinear Dyn.* **2013**, *73*, 1045–1054. [CrossRef]
29. Hirokami, T.; Maeda, Y.; Tsukada, H. Parameter estimation using simultaneous perturbation stochastic approximation. *Electr. Eng. Jpn.* **2006**, *154*, 30–39. [CrossRef]
30. Ozdemir, M.C.; Eggert, T.; Straube, A. Improving the repeatability of two-rate model parameter estimations by using autoencoder networks. *Prog. Brain Res.* **2019**, *249*, 189–194. [PubMed]
31. Olaizola, J.; Bouganis, C.S.; De, A. Real-time servo press force estimation based on dual particle filter. *IEEE Trans. Ind. Electron.* **2020**, *67*, 4088–4097. [CrossRef]
32. Kaelbling, L.; Littman, M.; Moore, A. Reinforcement learning: A survey. *J. Artif. Intell. Res.* **1996**, *4*, 237–285. [CrossRef]
33. Sutton, R.; Barto, A. *Reinforcement Learning: An Introduction*; The MIT Press: Cambridge, MA, USA; London, UK, 2005.
34. Farias, V.; Moallemi, C.; Van, B.; Weissman, T. Universal Reinforcement Learning. *IEEE Trans. Inf. Theory* **2010**, *56*, 2441–2454. [CrossRef]
35. Watkins, J.; Dayan, P. Q-learning. *Mach. Learn.* **1992**, *8*, 279–292. [CrossRef]
36. Vamvoudakis, K.G.; Lewis, F.L. Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica* **2010**, *46*, 878–888. [CrossRef]
37. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef] [PubMed]
38. Zhang, D.; Gao, Z.; Lin, Z. An online control approach for forging machine using reinforcement learning and taboo search. *IEEE Accesses* **2020**, *8*, 158666–158678. [CrossRef]